1 Formalizmus (10 bodov)

V tejto úlohe klaďte hlavný dôraz na presnosť a korektnosť práce s formalizmom z prednášky.

V tejto úlohe je povolené bez dôkazu použiť len nasledujúce skutočnosti:

- definíciu množiny primitívne rekurzívnych funkcií,
- primitívnu rekurzívnosť funkcií $add(x, y) = x + y$ a $mul(x, y) = x \cdot y$.

Vyriešte nasledujúce úlohy:

a) (4 body) Dokážte, že $zajac(x, y) = x + 2y + 47$ je primitívne rekurzívna.

b) (6 bodov) Vyberte si jednu z nasledujúcich dvoch možností:
b1) O funkcii $tri(x, y, z) = 3$ dokážte, že je primitívne rekurzívna.
b2) Za predpokladu, že platí b1, dokážte, že $pow3(x) = 3^x$ je primitívne rekurzívna.

(Kompozíciu zapisujte zľava doprava, ako ja na prednáškach. Teda napr. pre unárne funkcie f a g funkcia $Comp[f, g]$ je funkcia h definovaná tak, že $\forall x : h(x) = g(f(x))$ – teda „zoberiem vstup, najskôr spravím f, potom spravím g".)

2 Formalizmus (10 bodov) – riešenie

a) Vyrobme $f_0 \equiv Comp[add, P_1^2, add]$. Keďže $(x + 2y) = (x + y) + y$, vidíme, že $f_0(x, y) = x + 2y$.
Teraz $\forall i$ definujme $f_{i+1} \equiv Comp[s, f_i]$. Zjavne $f_{47} \equiv zajac$.

b1) Z nulárnej nuly z vyrobíme unárnu nulu pomocou primitívnej rekurzie: $z_1 \equiv PR[z, P_1^2]$.
Z nej si už vieme vyrobiť nulu ľubovoľnej arity aj kompozíciou, napr. $z_3 \equiv Comp[P_1^3, z_1]$.
A odtiaľ je už ľahké vyrobiť $tri \equiv Comp[s, Comp[s, Comp[s, z_3]]]$.

b2) Funkciu $pow3$ priamo vyrobíme primitívnou rekurziou.
Pre $x = 0$ nám výstup povie funkcia $j \equiv Comp[z, s]$.
Indukčný krok spraví binárna funkcia, ktorá svoj prvý argument vynásobí tromi.
Tú si vyrobíme napr. takto: $k \equiv Comp[Comp[P_1^2, P_1^2, P_1^2, tri], P_1^2, mul]$.
(Slovne: do mul dosadíme binárnu konštantu 3 a prvý vstup, čím dostaneme funkciu, ktorá vynásobí svoj prvý vstup tromi.)
Teraz teda $pow3 \equiv PR[j, k]$.

b2) Iné riešenie:
Z unárnej jednotky j_1 (vznikne analogicky ako tri) a mul vieme primitívnou rekurziou vyrobiť umocňovanie: $pow \equiv PR[j_1, Comp[P_1^3, P_3^3, mul]]$.
A do neho už len dosadíme: $pow3 \equiv Comp[P_1^2, t_2, pow]$, kde t_2 je binárna konštanta 3.
(Teda napr. $t_2 = Comp[P_1^2, P_1^2, P_1^2, tri]$.)

3 Známa pôda (10 bodov)

V tejto úlohe sa pýtam na priamo odprednášané veci, ukážte, že im dostatočne rozumiete.

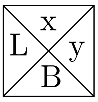
Pripomeňme si, že dlaždicový program (Wang tile program) je štvorica (množina typov dlaždíc, farba ľavej steny, farba pravej steny, farba spodnej steny). Takýto program rozpoznáva jazyk tých slov, pre ktoré sa dá korektne vydláždiť vhodný obdĺžnik.

Formálne zostrojte dlaždicový program, ktorý bude rozpoznávať jazyk desiatkových zápisov prirodzených čísel, ktoré nie sú deliteľné siedmimi. Slovne zdôvodnite, prečo vaša konštrukcia funguje.

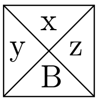
Priestorová zložitosť vášho riešenia (t. j. funkcia udávajúca počet riadkov, ktoré dláždenie potrebuje, v závislosti od dĺžky vstupu) musí byť asymptoticky optimálna. Dokážte to o nej.

4 Známa pôda (10 bodov) – riešenie

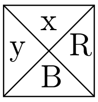
Jazyk zo zadania je regulárny. Vieme ho rozpoznávať konečným automatom, ktorý zľava doprava číta vstupné slovo a v stave si pamätá zvyšok, ktorý dávala doteraz prečítaná časť slova po delení siedmimi. Tento automat vieme ľahko prerobiť na dlaždicový program (D, L, R, B), kde D obsahuje nasledujúce dlaždice:



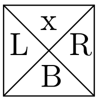
$x \in \{1, \dots, 9\}$; $y = x \bmod 7$



$x \in \{0, \dots, 9\}$; $y \in \{0, \dots, 6\}$; $z = (10y + x) \bmod 7$



$x \in \{0, \dots, 9\}$; $y \in \{0, \dots, 6\}$; $(10y + x) \bmod 7 \neq 0$



$x \in \{1, 2, 3, 4, 5, 6, 8, 9\}$

K ľubovoľnému slovu z požadovaného jazyka existuje jednoriadkové dláždenie a to je zjavne optimálne.

5 Exkurzia do neznáma (10 bodov)

V tejto úlohe sa pýtam na niečo, čo som neprednášal, a chcem vidieť, ako si s tým poradíte.

2D unárny deterministický Turingov stroj počítajúci čiastočnú funkciu (2DuDTSpčf, skrátene tiež Ignác) je štvorica (K, δ, q_0, F), významy jednotlivých zložiek sú rovnaké ako u klasického TS.

Pracovný priestor je prvý kvadrant roviny, rozdelený na štvorcové políčka a olemovaný zarážkami, aby nám Ignác z neho nevypadol. Presnejšie, na políčkach so súradnicami $(x, -1)$ a $(-1, x)$ sú zarážky. V každom okamihu sa Ignác nachádza na jednom z políčok. Pracovná abeceda je vždy $\{a\}$. Každý krok výpočtu vyzerá nasledovne:

- Ignác prečíta políčko, kde práve stojí.
  (T. j. zistí, či je na ňom a, blank, alebo zarážka značiaca, že už vyšiel z vymedzenej plochy.)
- Ignác zapíše a (ak nestojí na zarážke).
- Podľa riadku δ-fcie zodpovedajúceho aktuálnemu stavu a tomu, čo práve prečítal, Ignác zmení stav a pohne sa jedným z 8 základných smerov. (Ak stojí na zarážke, má menej ako 8 možností, kam ísť.)

Na začiatku sú všetky políčka s nezápornými súradnicami prázdne. Výpočet hodnoty $f(n)$ sa začína tým, že do políčok $(0, 0)$ až $(n - 1, 0)$ zapíšeme znak a, položíme Ignáca na políčko $(0, 0)$, nastavíme jeho stav na q_0 a spustíme ho.

Dodefinujte, kedy Ignácov výpočet končí a ako v tom okamihu z jeho konfigurácie získame výsledok výpočtu. A to nie hocijako – spravte to tak, aby výsledný model bol Turing complete. Teda trebárs ku každej funkcii v našom zjednodušenom Pascale musí existovať Ignác, ktorý počíta to isté.

A nezabudnite na dôkaz.

6 Exkurzia do neznáma (10 bodov) – riešenie

Bolestivé riešenie: Simulujeme nejaký prepisovací systém. Symboly jeho abecedy si môžeme napr. reprezentovať ako rôzne vysoké stĺpčeky a čok. Vieme prečítať symbol, zapamätať si ho v stave, ale prečítaním ho v princípe z pásky stratíme. To sa dá zachrániť tak, že novú konfiguráciu vždy vyrobíme napravo od predchádzajúcej.

Presná formulácia takéhoto riešenia zahŕňa nechutne veľa technických detailov.

Lepšie riešenie: Začneme tým, že prídeme na koniec vstupného slova, čím nastavíme hlavu na pozíciu $(n, 0)$. Od tohto okamihu ďalej používame pozíciu na páske ako ekvivalent dvoch registrov Minského registrového stroja; stav Ignáca nám predstavuje číslo aktuálne simulovanej inštrukcie. Obsah pásky spokojne ignorujeme.

(Výstupnú hodnotu následne teda definujeme ako x-ovú súradnicu hlavy v okamihu, kedy Ignác prvýkrát dosiahne akceptačný stav.)